BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS:

1. (currently amended) A stub search loading system for, in executing remote method invocation from a plurality of clients to a server, downloading a stub necessary in a request source client from the server, wherein

the request source client comprises stub search means for sending a stub request formed from a stub name and client identifier to the server and receiving a stub returned from the server, and

the server comprises a stub search interface for, in response to the stub request from the request source client, returning to the request source client the stub appropriate for a runtime environment of the request source client on the basis of the designated stub name and client identifier, wherein,

the server supports a plurality of different runtime environments and returns to the request source client the stub selected for the runtime environment of the request source client, and

the stub search interface of the server utilizes the client identifier to determine the appropriate runtime

environment for the request source client from the plurality of different runtime environments supported by the server.

2. (currently amended) A system according to claim 1, wherein

the server further comprises a ~~stub set in which a stub~~ plurality of stub sets, each of the stub sets corresponding to one of the plural runtime environments and including stubs to be used together with a skeleton used in the server at the time of remote method invocation from the request source client ~~is prepared for each of types of the clients having different runtime environments~~, and

upon receiving the stub request from the request source client, said stub search interface searches said stub set for the corresponding stub on the basis of the designated stub name and client identifier and returns the stub to the request source client.

3. (original) A system according to claim 1, wherein

the server further comprises stub generation means (30) for generating, for each of types of the clients having different runtime environments, a stub to be used together with a skeleton used in the server at the time of remote method invocation from the request source client, and

upon receiving the stub request from the client, said stub search interface returns to the request source client the stub appropriate for the runtime environment of the

request source client, which is generated by said stub generation means on the basis of the designated stub name and client identifier.

4. (original) A system according to claim 2, wherein

the server further comprises stub generation means (30) for generating, for each of types of the clients having different runtime environments, a stub to be used together with a skeleton used in the server at the time of remote method invocation from the client, and

when the corresponding stub is not present in said stub set, said stub search interface returns to the request source client the stub appropriate for the runtime environment of the request source client, which is generated by said stub generation means on the basis of the designated stub name and client identifier.

5. (currently amended) A stub search loading method of, in executing remote method invocation from a plurality of clients to a server, downloading a stub necessary in a request source client from the server, comprising by comprising the steps of:

sending a stub request formed from a stub name and client runtime identifier from the request source client to the server;

upon receiving the stub request, returning from the server to the request source client a stub to be used together

with a skeleton used in the server at the time of remote method invocation from the request source client, the stub returned to the requesting source client being selected on the basis of the designated stub name and the client runtime identifier so that the stub is appropriate for a runtime environment of the requesting source client; and

receiving, at the request source client, the stub transmitted from the server.

6. (currently amended) A method according to claim 5, wherein the return step comprises the steps of

in the server on the basis of the designated client runtime identifier, selecting one of plural stub sets in which a stub to be used together with a skeleton used in the server at the time of remote method invocation from the request source client is prepared for each of types of the clients having different runtime environments,

selecting the stub having the designated stub name from the selected stub set, and

transmitting the selected stub from the server to the request source client.

7. (currently amended) A method according to claim 5, wherein the return step comprises the steps of

in the server on the basis of the designated stub name and client runtime identifier, generating a stub to be used together with a skeleton used in the server at the time

of remote method invocation from the request source client, and

transmitting the generated stub from the server to the request source client.

8. (currently amended) A method according to claim 6, wherein the return step comprises the steps of

in the server, when a stub cannot be selected from the stub set, generating a stub to be used together with a skeleton used in the server at the time of remote method invocation from the request source client on the basis of the designated stub name and client runtime identifier, and

transmitting the generated stub from the server to the request source client.

9. (currently amended) A server apparatus for providing a stub necessary in executing remote method invocation to a request source client in response to a stub request from the client, comprising:

a stub search interface for, in response to the stub request from the request source client, returning to the request source client the stub appropriate for a runtime environment of the request source client, the stub appropriate for the runtime environment of the request source client being determined, from plural supported runtime environments, on the basis of the designated stub name and client identifier.

10. (currently amended) An apparatus according to claim 9, wherein

said apparatus further comprises a ~~stub set in which a stub~~ plurality of stub sets, each of the stub sets corresponding to one of the plural runtime environments and including stubs to be used together with a skeleton used at the time of remote method invocation from the request source client ~~is prepared for each of types of the clients having different runtime environments~~, including the request source client, and

upon receiving from the request source client the stub request formed from the stub name and client identifier, said stub search interface searches said stub set for the appropriate stub on the basis of the designated stub name and client identifier and returns the stub to the request source client.

11. (currently amended) An apparatus according to claim 9, wherein

said apparatus further comprises stub generation means for generating, for each of plural types of clients having different runtime environments, including the request source client, a stub to be used together with a skeleton used at the time of remote method invocation from the request source client, and

upon receiving the stub request from the client, said stub search interface returns to the request source client the stub appropriate for the runtime environment of the request source client, which appropriate stub is generated by said stub generation means on the basis of the designated stub name and client identifier.

12. (currently amended) An apparatus according to claim 10, wherein

said apparatus further comprises stub generation means for generating, for each of plural types of clients having different runtime environments, including the request source client, a stub to be used together with a skeleton used at the time of remote method invocation from the client, and

when the corresponding stub is not present in said stub set, said stub search interface returns to the request source client the stub appropriate for the runtime environment of the request source client, which appropriate stub is generated by said stub generation means on the basis of the designated stub name and client identifier.

13. (currently amended) A client apparatus for downloading a stub necessary in a client in executing remote method invocation from a server, comprising:

stub search means for transmitting a stub request formed from a stub name and client identifier to a server having a stub search interface for, in response to the stub

request from the client, returning to the request source client the stub appropriate for a runtime environment of the request source client on the basis of the designated stub name and client identifier, and receiving the stub returned from the server apparatus in response to the stub request, the client identifier being used to determine the runtime environment of the request source client.

14. (currently amended) An apparatus according to claim 13, wherein

said apparatus further comprises a ~~stub set in which a stub~~ plurality of stub sets, each of the stub sets corresponding to one of the plural runtime environments and including stubs to be used together with a skeleton used in the server apparatus at the time of remote method invocation from the request source client ~~is prepared for each of types of the clients having different runtime environments~~, and

upon receiving from the request source client the stub request formed from the stub name and client identifier, the stub search interface searches said stub set for the appropriate stub on the basis of the designated stub name and client identifier and returns the stub to the request source client.

15. (original) An apparatus according to claim 13, wherein

said apparatus further comprises stub generation means for generating, for each of types of clients having different runtime environments, including the request source client, a stub to be used together with a skeleton used in the server apparatus at the time of remote method invocation from the request source client, and

upon receiving the stub request from the client, the stub search interface returns to the request source client the stub appropriate for the runtime environment of the request source client, which is generated by said stub generation means on the basis of the designated stub name and client identifier.

16. (original) An apparatus according to claim 14, wherein

said apparatus further comprises stub generation means for generating, for each of types of clients having different runtime environments a stub to be used together with a skeleton used in the server apparatus at the time of remote method invocation from the client, and

when the corresponding stub is not present in said stub set, the stub search interface returns to the request source client the stub appropriate for the runtime environment of the request source client, which is generated by said stub generation means on the basis of the designated stub name and client identifier.

17. (currently amended) A computer-readable recording medium which stores a program for executing stub search loading processing of, in executing remote method invocation from a plurality of clients to a server, downloading a stub necessary in a request source client from the server, wherein the program comprises:

a procedure code for sending a stub request formed from a stub name and client identifier from the request source client to the server, the client identifier being usable to determine a runtime environment type of the request source client;

a procedure code for, upon receiving the stub request, returning from the server to the request source client a stub to be used together with a skeleton used in the server at the time of remote method invocation from the request source client, on the basis of the designated stub name and client identifier, the stub being appropriate for the runtime environment type of the request source client as determined from the client identifier; and

a procedure code for receiving, at the request source client, the stub transmitted from the server.

18. (currently amended) A medium according to claim 17, wherein the program for executing the procedure code for returning comprises

a procedure code for, in the server on the basis of the designated client identifier, selecting one of plural stub sets, in which a stub to be used together with a skeleton used in the server at the time of remote method invocation from the request source client is prepared for each of types of the clients having different runtime environments,

a procedure code for selecting the stub having the designated stub name from the selected stub set, and

a procedure code for transmitting the selected stub from the server to the request source client.

19. (original) A medium according to claim 17, wherein the program for executing the procedure code for returning comprises

a procedure code for, in the server on the basis of the designated stub name and client identifier, generating a stub to be used together with a skeleton used in the server at the time of remote method invocation from the request source client, and

a procedure code for transmitting the generated stub from the server to the request source client.

20. (original) A medium according to claim 18, wherein the program for executing the procedure code for returning comprises

a procedure code for, in the server, when a stub cannot be selected from the stub set, generating a stub to be

used together with a skeleton used in the server at the time of remote method invocation from the request source client on the basis of the designated stub name and client identifier, and

a procedure code for transmitting the generated stub from the server to the request source client.

21. (currently amended) A computer-readable recording medium which stores a program for executing processing of providing a stub necessary in executing remote method invocation to a request source client in response to a stub request from the client, wherein the program comprises:

a procedure code for, in response to the stub request from the request source client, returning to the request source client the stub appropriate for a runtime environment of the request source client on the basis of the designated stub name and client identifier, the client identifier being used to determine the runtime environment of the request source client.

22. (currently amended) A medium according to claim 21, wherein the program for executing the procedure code for returning comprises

a procedure code for, in the server on the basis of the designated client identifier, selecting one of plural stub sets in which a stub to be used together with a skeleton used in the server at the time of remote method invocation from the

request source client is prepared for each of types of the clients having different runtime environments,

a procedure code for selecting the stub having the designated stub name from the selected stub set, and

a procedure code for transmitting the selected stub from the server to the request source client.

23. (original) A medium according to claim 21, wherein the program for executing the procedure code for returning comprises

a procedure code for, in the server on the basis of the designated stub name and client identifier, generating a stub to be used together with a skeleton used in the server at the time of remote method invocation from the request source client, and

a procedure code for transmitting the generated stub from the server to the request source client.

24. (original) A medium according to claim 22, wherein the program for executing the procedure code for returning comprises

a procedure code for, in the server, when a stub cannot be selected from the stub set, generating a stub to be used together with a skeleton used in the server at the time of remote method invocation from the request source client on the basis of the designated stub name and client identifier, and

a procedure code for transmitting the generated stub from the server to the request source client.

25. (new) A stub search loading system, comprising:

a stub search section for installation on plural client computers of differing runtime environments; and

a stub search interface for installation on a server computer in communication with the plural client computers,

the stub search section permitting each client computer, as a requesting client computer, to issue a stub class request to the stub search interface, and to download from the server computer the requested stub class for remote method invocation to the server computer, the downloaded stub class being provided from the server computer to the requesting client computer in the runtime environment of the requesting client computer,

the stub class request comprising a stub class name and a client identifier, the runtime environment of the requesting client computer being determinable from the client identifier,

the stub search interface responding to the stub class request, on the basis of the designated stub class name and client identifier, by returning to the requesting client computer the requested stub class appropriate for the runtime environment of the requesting client computer.

26. (new) A stub search loading system of claim 25, wherein, the stub search interface responds to the stub class request by:

using the client identifier to determine which of a plurality of stub class sets for the differing runtime environments is appropriate for searching to determine if the server computer has an existing stub class that matches the stub class name in the stub class set of the runtime environment of the requesting client computer, and

when the server computer does not have the existing stub class that matches the stub class name in the stub class set of the runtime environment of the requesting client computer, the server computer dynamically generates an appropriate stub class, in the runtime environment of the requesting client computer, for returning to the requesting client computer.